### **Название задачи: Открытие депозитов онлайн**

### **Автор: lppt**

### **Дата: 2025-01-06**

### **Функциональные требования**

Опишите здесь верхнеуровневые Use Cases. Их нужно оформить в виде таблицы с пошаговым описанием:

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Действующие лица или системы** | **Use Case** | **Описание** |
| 1 | Клиент, Сайт, АБС, Менеджер бэк офиса | Получение списка доступных депозитов | 1. Менеджер бэк офиса в АБС запускает процесс выгрузки ставок и доступных депозитов для сайта.  2. Клиент заходит на сайт, переходит на страницу депозитов.  3. Сайт возвращает список доступных депозитов с актуальными ставками |
| 2 | Клиент, Сайт, АБС, Менеджер кол- центра, Фронт офис | Отправка заявки на депозит через сайт | 1. Клиент на сайте выбирает доступный депозит, вводит номер телефона, ФИО и подает заявку  2. Сайт отправляет заявку в систему кол-центра  3. Менеджер кол-центра звонит клиенту, предлагает особые условия  4. Менеджер заводить заявку в АБС с указанием условий  5. Клиент приходит в отделение  6. Фронт офис идентифицирует пользователя и открывает депозит |
| 3 | Клиент, интернет банк, АБС, Менеджер бэк офиса | Формирование персонализировнных ставок для интернет-банка | 1. Менеджер бэк офиса в АБС запускает процесс выгрузки персональных ставок для интернет банка  2. Пользователь заходит в интернет банк, получает персональные предложения |
| 4 | Клиент, интернет-банк, смс-шлюз, АБС, Менеджер бэк офиса | Отправка заявки на депозит через интернет-банк | 1. Клиент в интернет-банке указывает счет, сумму депозита  2. Интернет-банк формирует проверочный код и отправляет его через смс-шлюз.  3. Пользователь вводит проверочный код в интернет банк  4. Интернет банк отправляет заявку в АБС  5. Менеджер бэк офиса обрабатывает заявку  6. АБС отправляет нотификацию в смс шлюз |

### **Нефункциональные требования**

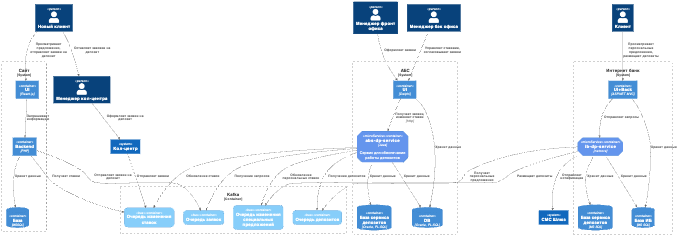
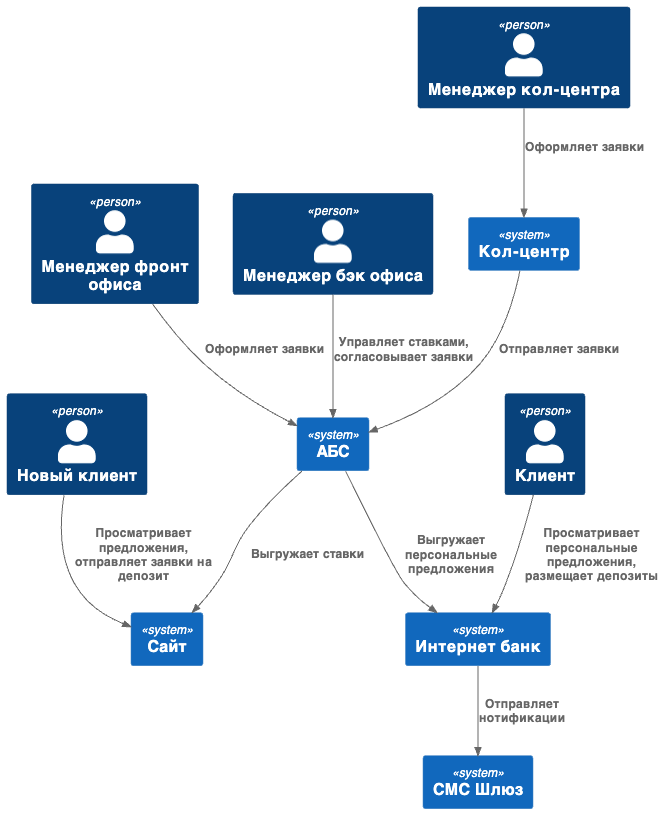
Опишите здесь нефункциональные требования и архитектурно-значимые требования.

|  |  |
| --- | --- |
| **№** | **Требование** |
| 1 | Работа 24/7 |
| 2 | Доступность 99.9 |
| 3 | Отклик по операциям - миллисекунды |
| 4 | Шифрование данных |
| 5 | Используем технологии, используемые в банке |
| 6 | Используем дизайн систему |
| 7 | Делаем систему, масштабируемую на 2 ЦОД |

### **Решение**

Приведите диаграммы контекста и контейнеров в модели C4. Опишите там основные компоненты и интеграции всех элементов решения.

Также опишите, какой логикой вы руководствовались в ходе принятия решений и выбора технологий. Не забывайте, что необходимо учесть все функциональные и нефункциональные требования.



Для обеспечения новых процессов и обеспечения их отказоустойчивости, а так же интеграции всех систем между собой предлагается шину. Так как команда банка считает что самым перспективным вариантом будет kafka, то возьмем ее.

Через шину будем прогонять все заявки на депозиты, а так же процесс изменения ставок.

**Поддержка процессов в Интернет банке**

В виду того что команда разработки банка утверждает что текущая реализация не может поддерживать взаимодействие в kafka, то предлагается реализовать новый микросервис в контексте интернет банка ib-dp-service

Реализовывать его предлгается на netcore, в виду того что весь ИБ написан на .net framework и команда знакома с этой средой.

В его функции будет входить

- Хранение и отдача предложений по депозитам

- Получение запросов на депозиты от интернет банка

- Работа с смс шлюзом

- Сохранение результатов выполнения процесса депозитов

**Микросервис депозитов в АБС**

Так как команда АБС говорит что текущая реализация не позволяет масштабироваться, для работы с депозитами на уровне АБС предлагается реализовать новый микросервис для обеспечания всех новых процессов и взаимодействия с kafka в контексте АБС abs-dp-service.

Для реализации предпочтительно будет выбрать Java в виду того что у команды АБС есть опыт.

В задачи сервиса будет входить

- Прием заявок на депозиты от менеджеров кол центра

- Прием заявок с сайта

- Прием заявок от ИБ

- Процесс обновления ставок

- Процесс расчета персональных ставок

- Обеспечение работоспособности workflow по депозитам

Для всех операций которые можно отделить от монолитной бд, желательно использовать либо отдельную бд, либо хотя бы отдельную схему

### **Альтернативы**

1. Использование дельфи на abs-dp-service.

С одной стороны команда лучше знакома с технологией, с другой в перспективе развития процесса проще будет искать разработчиков на java

2. Добавление функционала abs-dp-service в монолит.

Это позволит ускорить процесс разработки, но скорее всего вызовет проблемы с масштабирование процесса. В это решение можно перейти, если оценка разработки отдельного сервиса выйдет за рамки бюджета.

**Недостатки, ограничения, риски**

Подробно опишите здесь недостатки, ограничения и риски выбранного решения.

- Возможны сложности в работе дельфи приложения с новым сервисом, т.к. помимо доступа к бд придется сверлить новые дырки в фаерволах для всех установок.

- Необходимо обеспечить отказоустойчивость kafka, т.к. он становится важной частью всего процесса

- Отказ интернет банка может повлечь остановку всего процесса. Не смотря на то что мы сняли с него большую часть нагрзуки, все равного есть зависимость.

- Oracle и MS SQL ушли из России, так что поддержки не будет. Желательно подумать о переходе на PostrgreSQL, хотя бы в новых сервисах